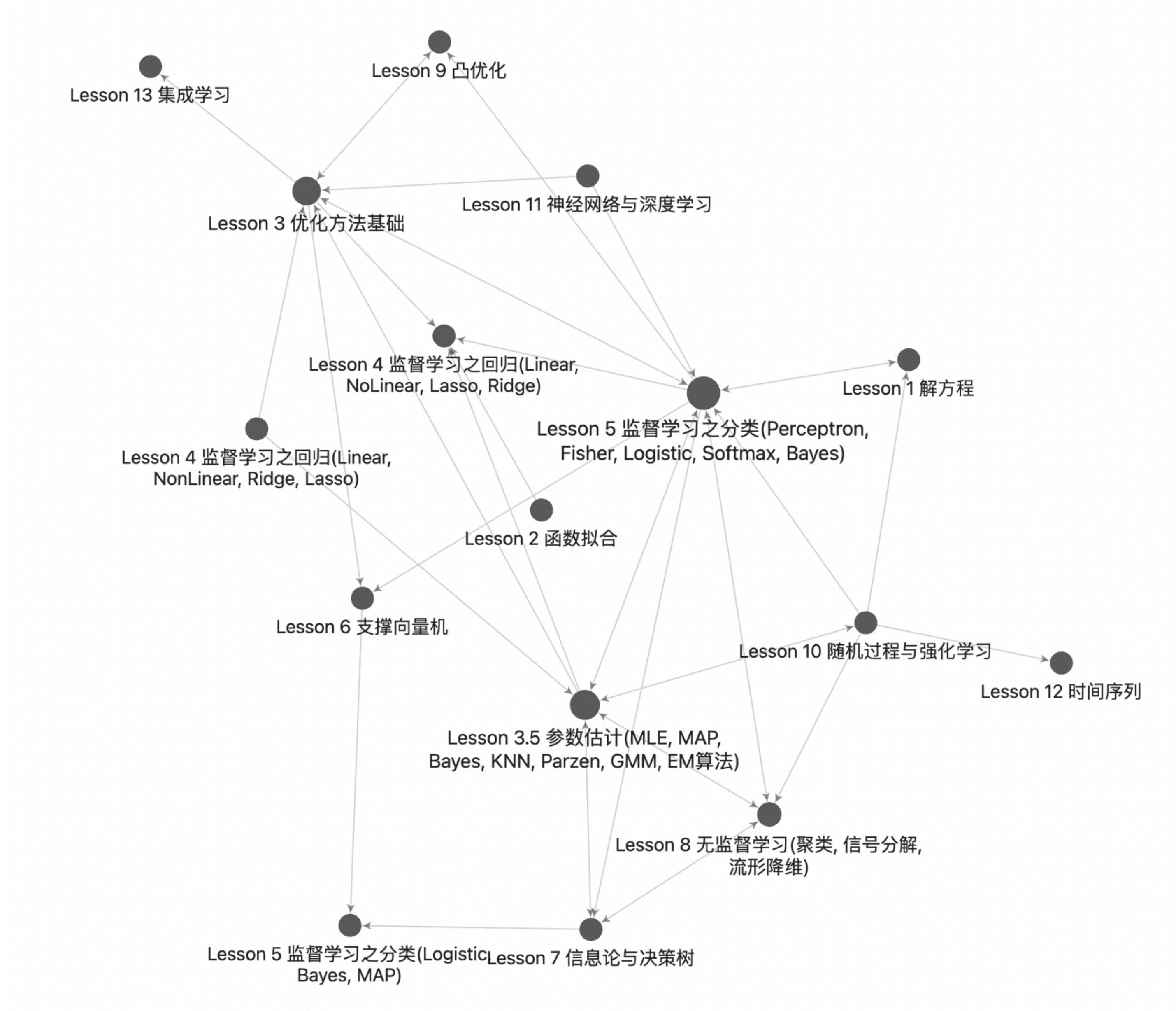
机器学习：解方程(Equation Solving)

~~Copyright: Jingmin Wei, Automation - Pattern Recognition and Intelligent System, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology~~

Copyright: Jingmin Wei, Computer Science - Artificial Intelligence, Department of Computer Science, Viterbi School of Engineering, University of Southern California

---



在华科 A 院学习机器学习两年多，却被在 USC 的机器学习课彻底打败，本科课堂也唯有曹老板的模式识别前短短四周能与之抗衡，除了感慨教育质量之差距，也感慨 ML 理论世界之魅力。$Vatsal$ 教授上课徒手推公式，生动形象阐述每一个算法，理论完整，逻辑清晰(不愧是斯坦福博士 MIT 博后，膜)，让自己感觉好像还是个刚入门的萌新。个人以为，ML 理论真的很重要。大家都说深度学习入门简单，看个代码调个参，调个包，不断做消融实验就能出结果，可是当真正涉及模型改进的时候，这些基础理论以及优化算法，才是能帮助到你的知识。

1. 数据科学基础课堂&课后笔记的最佳食用方法

该笔记的主体是 AIA 数据科学基础的课堂笔记，完全是个人下课后手打的，一开始只是想完成这门课的大作业，也就是**每个人都要抄老师规定部分的课堂笔记**。后来觉得袁老师讲得太好了，所以每堂课就全部记下来了。之后刷完了一本 400 多页的数学书，闲着无聊把西瓜书挑着看了一遍，然后又在 USC 上了 老师的[机器学习]课，索性把笔记从头到尾优化了一遍。之后我会把这部分笔记和 USC 上机器学习课记的英文笔记一并上传 $github$ 。

主要参考资料有：数据科学基础课堂和课后笔记(主体)，西瓜书，《机器学习中的数学》，模式识别部分课件，一些经典的论文，$csdn$ 上的一些大佬的文章。

尤其要感谢一下花姐当时每天上课帮我扫描 pdf 。因为老师不准外传笔记的原因，所以笔记和板书我都是上课的时候拍下来或者拜托花姐帮我拍，回头每天上完课花 3 小时左右全部补上。袁烨老师的板书写得非常好，所以他的所有内容我都抄下来了。有时候其他的老师会用课件讲课，课后我也把每页课件都补上了。

首先简单说下数据科学的定义。$Data\ Science$ 在国外，尤其是在美国，是需要和具体内容和知识领域结合的，比如说 $DS\ in\ health, DS\ in\ Enviroment\ Engineering, DS\ and\ Analytics$ ，可见数据科学是一个广义上的概念，也有专门数据科学的硕士学位。而这门课的内容，更准确来说，是在讲机器学习，或者说是机器学习中的数学推导。其实如果大家的 $Prerequisite$ 有模式识别或者机器学习，上这门课会更好理解。但是考虑到大家没上模式识别，所以我从开学开始一直在完善笔记的内容，争取让很多算法的入门门槛变得更低。

$ML\ From\ Probability\ Perspective$ ，也就是从概率论角度理解机器学习，其实是最合适的。但是本科的模式识别这门课，其实更多是从梯度下降的优化算法讲起，**重点是带大家理解"学习"**。而数据科学基础这门课，又是从一些基本的计算方法开始讲起的，并没有详细讲梯度下降，而是着重于不同的优化方法，也掺杂了一元多元微积分，矩阵，概率论的一些知识，**重点是带大家理解"优化"**。所以为了方便大家理解一些重要算法，我加了**第 3.5 章：概率密度函数的估计**，概率论中的最大似然，最大后验，贝叶斯这三个思路非常重要，也能帮大家更好的理解之后的线性回归，正则化，逻辑回归等一些经典算法的原理，同时也帮大家理解机器学习从概率角度出发，到底需要完成一个怎样的任务。

最后列一下每章的概要和重点。有些是老师上课没讲的，完全是我课后补充的内容，我也列出来了，大家这部分的算法可以根据需求自己选看。

第 1,2 章是一些基础的解方程和函数拟合问题，这里我没有额外补充，这两个部分也是我被袁烨老师圈粉的部分，讲得深入浅出，后面就开始蒙了。

第 3 章是袁老师教的，讲一些基本的优化方法。其中**黑塞矩阵，极值判别法则等等数学部分都很重要。在一阶优化算法的部分，除了梯度下降算法，其他算法都是我自己后面加上去的，其中一阶牛顿法我主要是参考的计算方法的课件，老师上课并没有讲随机梯度下降，这一部分的概念很重要且非常容易混淆，且是后面多个算法优化的重要内容，推荐大家认真看下 SGD 及其变种**，当时我刚好在写 $PyTorch$ 优化器的源码分析，顺便就把多个不同的优化算法的内容写上去了。后面的二阶优化方法和分治法，也都是我之后补充上去的。

第 3.5 章参数估计，全部内容都是我之后加的，**是我觉得很重要的知识点**，不夸张的说，最大似然估计几乎贯穿了整个机器学习的理论推导。数据科学的课堂上并没有这部分内容，而是把这部分穿插在了 4,5 两章来讲解，但是模式识别的老师讲了(虽然讲的顺序我也觉得有问题)。原则上是选看，但是我觉得不看 3.5 章理解 4,5 两章可能会存在一定困难，**尤其是经典的抛硬币问题，非常推荐大家在老师讲基本的回归分类算法前看一下**。变分推断会比较难，看不懂没关系，因为我也不太懂... GMM, EM 也是非常经典的参数估计方法，值得一看。

第 4 章是袁老师教的，讲基本的回归算法。**其中，线性回归的梯度下降优化方法是我加上去的，这一部分很重要，之后的所有随机梯度下降的算法阐述将沿用这一框架。因为老师当时重点是从最大似然和最大后验角度出发讲优化算法，随机梯度下降是我之后补上去的。

第 5 章讲基本的分类算法，袁老师上课讲了 $Logistic$ ，$Bayes$ 是程骋老师教的，但是这两个部分也有很多没讲清楚。这一章其中的大部分都是我后期加上去的：**感知机，线性判别分析，以及 $Softmax$ 回归，这些是我认为比较基础但是也很重要的分类算法**。逻辑回归部分，我补充了 $Sigmoid$ 函数的推导，损失函数的梯度计算，分治法的优化；$Bayes$ 我补充了高斯贝叶斯模型部分。

第 6 章是袁老师教的，支撑向量机也是我觉得最难的一章之一，我也补充了非常多的内容。添加了二次规划求解，软间隔问题，分治法，支撑向量的判断等，同时补充了核方法的部分。这里是我当时完全听不懂的一章，我本来以为模式识别我学的很明白了，上了数据科学才发现自己只学懂了合页损失和 KKT 条件。**这一章的拉格朗日对偶方法是一个非常重要的优化算法，大家可以多看看。**

第 7 章是程老师教的，主要是讲信息论和决策树。这里我把信息论部分的数学公式和一些熵的性质都补充完整了，并添加了 KL, JS 散度部分。这里程老师上课的顺序有点问题，是先讲的决策树然后再讲的信息论，其实**先学懂信息论基础再看决策树**会简单很多，所以我调整了一下顺序，同时我也把第 7,8 章的上课顺序调换了一下。

第 8 章主要讲无监督学习中的几个常用算法，袁老师教的局部线性嵌入 LLE 和 $K-Means$，程老师教的 PCA 。袁老师的板书写得很好，但是局部线性嵌入的一堆公式根本听不懂。我补充了 $Mean-Shift$ 聚类算法，基于 PCA 的优化目标，补充了矩阵的特征值和奇异值分解计算。流形学习部分，除了局部线性嵌入 LLE ，我补充了随机近邻嵌入 SNE 和谱嵌入 LE 算法。**无监督学习很重要，但是算法的数学推导其实挺难的，实在看不懂不用勉强。**

第 9 章是袁老师当时请了东北大学一个控制系的老师来教的。说是 90 分钟讲清楚凸优化，讲到后面我除了凸集之外啥都没听懂，后面还搁那介绍自己实验室。所以这里我只补充了一些凸优化问题的基本概念，相关知识的深入大家可以自行查阅凸优化的资料。

第 10 章是袁老师教的，主要是讲强化学习。当时上课的时候也是不知所云，袁老师上课的板书还是很好看的，缺点就是我啥也听不懂。所以为了方便大家理解，这章我补充了一些随机过程里的东西，包括马尔科夫过程，隐马尔科夫模型等等，然后把强化学习的笔记部分补充了一下。

第 11 章就不多说了，程老师教的，介绍了一些神经网络的基本概念。详细的可以看我的专栏 。其中，老师上课没讲过的半监督学习算法在专栏里的图卷积网络部分，有兴趣可以了解一下。

第 12 章没有额外补充，当时是岳作功老师教的，讲得还是很清晰的。

第 13 章也是岳老师教的，讲得也挺好的。我只补充了基于分治法的 $AdaBoost$ 优化算法推导和 $Stacking$ 算法。岳老师上课没讲 $Stacking$ ，我觉得这个理解很简单，就是拿第一级强分类器的概率图输出作为第二级弱分类器的输入，缓解过拟合，所以我也没过多补充。

说一下我认为的重点章节：3, 3.5, 4, 5, 6, 8，这些之后模式识别还会再讲一遍。当然，第 11 章也重要但是这部分当时是程老师念的 $Slide$，神经网络的理论和最佳实践更看大家自己去学习和总结。

笔记的最佳食用方式已经留给大家了，充分使用看个人。

2. 解方程

从解方程开始：$Ax=b$

$$\begin{cases} x_1+x_2+x_3=1 \\ x_1+2x_4+x_3=-1 \\ x_1+3x_2+9x_3=1 \end{cases}$$

$$令A=\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{bmatrix}, \quad X=\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad b=\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

如下的很多算法，在本科的计算方法的课程中也有学到。

3. 高斯消元法

$$
\begin{aligned}
[A|\,b] &= \begin{bmatrix} 1 & 1 & 1 & & 1 \\ 1 & 2 & 4 & & -1 \\ 1 & 3 & 9 & & 1 \end{bmatrix} \\
&= \begin{bmatrix} 1 & 1 & 1 & & 1 \\ 0 & 1 & 3 & & -2 \\ 0 & 0 & 1 & & 2 \end{bmatrix}
\end{aligned}
$$

$x_3 = 2, \quad x_2 = -8, \quad x_1 = 7$。$\dim(A) = N$，计算复杂度 $O(N^3)$

4. LU 分解法

$$
\begin{aligned}
A = LU \to &\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \\
= &\begin{pmatrix} 1 & 0 & 0 \\ m_{21} & 1 & 0 \\ m_{31} & m_{32} & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{pmatrix}
\end{aligned}
$$

$Ax = b$ 等价于 $LUx = b, Ux = y$

1.$Ly = b$

2.$Ux = y$

$$
\begin{aligned}
A = IA = &\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} \\
= &\begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{2} & 1 & 0 \\ -\frac{1}{4} & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ 0 & -2.5 & 4.5 \\ 0 & 1.25 & 6.25 \end{pmatrix} \\
= &\begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{2} & 1 & 0 \\ -\frac{1}{4} & \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ 0 & -2.5 & 4.5 \\ 0 & 0 & 8.5 \end{pmatrix} \\
= &L \cdot U
\end{aligned}
$$

5. 迭代法(高斯消元)

$$
\begin{cases} 4x - y + z = 7 \\ 4x - 8y + z = -21 \\ -2x + y + 5z = 15 \end{cases}
$$

$$x_{k+1}=\frac{7+y_k-z_k}{4}$$
$$y_{k+1}=\frac{21+4x_k+z_k}{8}$$
$$z_{k+1}=\frac{15+2x_k-y_k}{5}$$

算法：

1. 初始化 (x_0, y_0, z_0) 。
2. 重复 (*) 直至收敛。
3. $(x_{k+1}-x_k)^2+(y_{k+1}-y_k)^2<\varepsilon$

假设 $(x_0, y_0, z_0)=(1,2,2)$

$$\begin{array}{cccc} k & x_k & y_k & z_k \\ 0 & 1 & 2 & 2 \\ 1 & 1.75 & 3.375 & 3 \\ 2 & 1.84375 & 3.875 & 3.025 \\ \vdots & & & \\ 15 & 1.99999993 & 2.9999993 & 3.9999985 \end{array}$$

如果调整两个式子的顺序：

$$\begin{cases} -2x+y+5z=15 \\ 4x-y+z=7 \\ 4x-8y+z=-21 \end{cases}$$

$$x_{k+1}=\frac{15-y_k-5z_k}{2}$$
$$y_{k+1}=\frac{21+4x_k+z_k}{8}$$
$$z_{k+1}=7-4x_k+y_k$$

$$\begin{array}{cccc} k & x_k & y_k & z_k \\ 0 & 1 & 2 & 2 \\ 1 & -1.5 & 3.375 & 5 \\ 2 & 6.6875 & 2.5 & 16.375 \\ \vdots & & & \\ 15 & \infty & \infty & \infty \end{array}$$

发现不收敛。引入判断算法收敛的方式：对角占优矩阵。

$$A=\begin{pmatrix}4 & -8 & 1\\ 4 & -1 & 1\\ -2 & 1 & 5\end{pmatrix}$$
$$\text{换顺序后：}A=\begin{pmatrix}-2 & 1 & 5\\ 4 & -1 & 1\\ 4 & -8 & 1\end{pmatrix}$$

定义：对角占优矩阵

矩阵 A 为严格对角占优矩阵，当且仅当：

$$\forall k,\quad |a_{kk}|>\sum_{j=1,j\neq k}^{N}|a_{kj}|,\quad A\in |R^{N\times N}$$

若 A 为严格对角占优矩阵，则算法收敛。

证明：

$$Ax=b\Leftrightarrow x+(A-I)x=b$$
$$\text{令 }D=diag(A),\text{ 则 }Dx+(A-D)x=b$$
$$x_{k+1}=D^{-1}(D-A)x_k+D^{-1}b$$

6. 高斯 - $Sedel$ 法

$$\begin{cases}x_{k+1}=\frac{7+y_k-z_k}{4}\\ y_{k+1}=\frac{21+4x_{k+1}+z_k}{8}\\ z_{k+1}=\frac{15+2x_k-y_{k+1}}{5}\end{cases},\quad \text{加速计算}$$

7. 优化法

考虑二次型函数：

$$f(x)=\frac{1}{2}x^TAx-b^Tx+c$$

定义梯度：

$$\nabla f(x)=\begin{bmatrix}\frac{\partial f(x)}{\partial x_1}\\ \vdots\\ \frac{\partial f(x)}{\partial x_N}\end{bmatrix}$$

假设 A 是对称的：

$$\nabla f(x)=Ax-b$$

有引理：

$$\frac{\partial x^TAx}{\partial x}=(A+A^T)x$$
$$\frac{\partial b^Tx}{\partial x}=b$$

$concentrate:\nabla f(x)=0,\quad \min f(X)$ 。

$It's\ a\ critical\ point, only\ if$ A 是半正定矩阵，则 $\nabla f(x)=0$ 的解为 $\min f(x)$ 的最小值点。

定义： A 为半正定矩阵当且仅当 $\forall x\neq 0,\quad x^TAx\geq 0$ 。

半正定和正定矩阵的详细定义和应用在优化方法基础一章。

8. 特征值法

我们将在Lesson 5 监督学习之分类(Perceptron, Fisher, Logistic, Softmax, Bayes)的菲谢尔线性判别分析中更详细学习特征值和特征向量的应用。

定义：给定矩阵 $A\in|R^{N\times N}$ ，我们称 λ 为 A 的特征值， V 为对应的特征向量，当且仅当 λ,V 满足：

$$AV=\lambda V$$

$Ax=b$ ，假设 A 是可对角化矩阵，即存在可逆矩阵 V 和对角矩阵 D ，使得 $A=VDV^{-1}$。

$$VDV^{-1}=b$$

如果存在非零向量 x ，使得 $Ax=\lambda x$ 。则我们称 x 为 A 的特征向量，λ 为 A 的特征值。

$$Ax=\lambda x=\lambda Ix\Leftrightarrow(\lambda I-A)x=0$$

考虑一下两种情况：

1、$\lambda I-A$ 可逆

$$(\lambda I-A)^{-1}\cdot(\lambda I-A)x=0$$
$$x=0$$

2、$\lambda I-A$ 不可逆

$$\det(\lambda I-A)=0$$

举例：$A=\begin{pmatrix}1&3\\-1&5\end{pmatrix}$ 。

$$\det(\lambda I-A)=\det\begin{pmatrix}\lambda-1&-3\\1&\lambda-5\end{pmatrix}=(\lambda-2)(\lambda-4)$$

当 $\lambda=2$ 时，由得到 $\begin{pmatrix}1&-3\\1&-3\end{pmatrix}\begin{pmatrix}x_1\\x_2\end{pmatrix}=0$ 。可得 $x_1=3,\ x_2=1,$ 即$x=\begin{pmatrix}3\\1\end{pmatrix}$ 。

若求 $Ax=b$ ，令 $A=S\Lambda S^{-1}$ 。

其中 $S=(x_1,\cdots,x_n)$， $\Lambda=\begin{pmatrix}\lambda_1 & & \\ & \ddots & \\ & & \lambda_n\end{pmatrix}$。其中 $Ax_i=\lambda_i x_i$。

$$
\begin{aligned}
& AS=S\Lambda \\
\Leftrightarrow & A(x_1\cdots x_n)=(x_1\cdots x_n)\begin{pmatrix}\lambda_1 & & \\ & \ddots & \\ & & \lambda_n\end{pmatrix} \\
\Leftrightarrow & Ax_i=\lambda_i x+i
\end{aligned}
$$

$$
S\Lambda S^{-1}x=b \quad x^*=S^{-1}\Lambda^{-1}S
$$